

Fall 2010

# Derivation of Solar Insolation Estimates from LiDAR

Shane Grigsby  
*University of Colorado Boulder*

Follow this and additional works at: [http://scholar.colorado.edu/honr\\_theses](http://scholar.colorado.edu/honr_theses)

---

## Recommended Citation

Grigsby, Shane, "Derivation of Solar Insolation Estimates from LiDAR" (2010). *Undergraduate Honors Theses*. Paper 622.

This Thesis is brought to you for free and open access by Honors Program at CU Scholar. It has been accepted for inclusion in Undergraduate Honors Theses by an authorized administrator of CU Scholar. For more information, please contact [cuscholaradmin@colorado.edu](mailto:cuscholaradmin@colorado.edu).

# Derivation of Solar Insolation Estimates from LiDAR

Shane Grigsby

November 30, 2010

Honors Thesis Committee:

- Waleed Abdalati, Associate Professor of Geography, Committee Chair
- William Travis, Associate Professor of Geography, University of Colorado Honors Council Member
- Brian Muller, Assistant Professor of Planning and Design

## **Abstract**

This Honors Thesis describes the methodology that was used from May 2010 to December 2010 for deriving solar insolation estimates for the University of Colorado at Boulder campus from Light Detection And Ranging (LiDAR) data. Background is given on the LiDAR data set used, including acquisition considerations and the properties of the data set itself. The primary method used to derive solar insolation estimates of campus was the generation of first-return canopy Digital Elevation Models (DEMs) using ENVI, followed by slope and aspect calculations using the open-source Geographic Information System (GIS) GRASS. The slope and aspect raster tiles were used to derive solar insolation estimates using the *r.sun* GRASS module, and an extraction of campus building rooftops was accomplished using existing vector campus GIS data sets. The reasons and constraints that led to the development of this methodology are discussed and possible sources of error are considered. Finally, the findings and implications of this study are presented and additional steps to reduce error in future work are explored.

# Contents

<b>1</b>	<b>Introduction and Background</b>	<b>3</b>
1.1	Project Purpose and Funding . . . . .	3
1.2	Data Acquisition . . . . .	3
1.3	Project Resources . . . . .	5
<b>2</b>	<b>Methodology</b>	<b>6</b>
2.1	Development of a Work Flow . . . . .	6
2.2	Insolation Work Flow and Results . . . . .	7
2.2.1	Computational Compromises . . . . .	7
2.2.2	Basic inputs: Canopy DEMs, Rasters of Slope and Aspect . . . . .	8
2.2.3	Angle of Incidence, Shadows and Insolation Time . . . . .	8
2.2.4	Total, Direct Beam, Diffuse and Reflected Solar Insolation . . . . .	9
2.2.5	Final Processing . . . . .	11
<b>3</b>	<b>Sources of Error</b>	<b>11</b>
3.1	Accounted Sources of Error . . . . .	12
3.1.1	Atmospheric Optical Thickness . . . . .	12
3.1.2	Reflected and Diffuse Insolation Estimates . . . . .	13
3.1.3	Temporal Step Size . . . . .	13
3.2	Unaccounted Sources of Error . . . . .	14
3.2.1	Non-local Shading . . . . .	14
3.2.2	Clouds and Local Weather . . . . .	14
3.2.3	Tiling Effects . . . . .	14
<b>4</b>	<b>Concluding Remarks and Future Work</b>	<b>15</b>
4.1	Correction of Errors . . . . .	15
4.2	Other Work . . . . .	15

# 1 Introduction and Background

## 1.1 Project Purpose and Funding

In the spring of 2010, the Environmental Center funded the acquisition of a LiDAR data set of the University of Colorado at Boulder Campus. The data acquisition was funded as part of a Sustainable CU grant<sup>1</sup> to perform an alternative energy assessment in order to update the Blueprint for a Green Campus<sup>2</sup> carbon neutrality recommendations for on-site energy production. The goals of the data acquisition were to provide an up-to-date, three-dimensional data set of the campus at a high enough accuracy to derive solar insolation estimates, and also meet future development and environmental modeling needs of the CU Environmental Center. The proposal requested \$20,000 for data acquisition, modeling, and data set maintenance, and was later supplemented by a \$2,400 Undergraduate Research Opportunities Program<sup>3</sup> (UROP) grant to assist in the data processing and development phase.

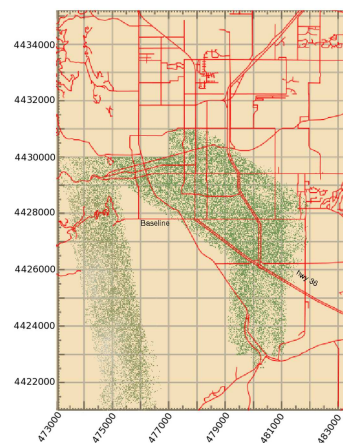
The purpose of the project was to provide solar modeling data to better inform university policy decision makers on future alternative energy investments in thermal and photovoltaic (PV) solar installations. The benefits to the researcher were an opportunity to become more familiar with LiDAR data acquisition and processing, as well as to gain experience in terrestrial raster based modeling. This thesis seeks to outline the project process from data acquisition through the development of an insolation modeling methodology, with a focus on sources of error and future best practices to minimize these errors. While the larger Environmental Center funded project is still ongoing, this thesis comes at the end of a first round of maximum and minimum bounding estimates, and aims to examine the merits and deficiencies that have been found in the particular methodologies used over the last six months of project work.

## 1.2 Data Acquisition

After securing funding for data acquisition, bids were solicited from Merrick<sup>4</sup>, the National Center for Airborne Laser Mapping<sup>5</sup> (NCALM) and Sanborn<sup>6</sup>. Sanborn declined to submit a bid as they had their LiDAR resources booked out for the remainder of the year. Merrick submitted a bid for a survey of three square kilometers at \$19,489 that included six-inch aerial photos as well as their proprietary software package MARS. NCALM had a data acquisition planned for the Critical Zone Observatory<sup>7</sup> (CZO) at the University's Mountain Research Station; they agreed to survey campus and the Flatirons for \$15,000 during their CZO Boulder Creek Watershed<sup>8</sup> campaign.

NCALM was chosen as a data provider for several reasons. As a for-profit firm, Merrick considered their processing techniques and work flow as confidential intellectual property. Since NCALM operated at-cost to support research, they were much more forthcoming with their work flow

Figure 1: Data Extent



<sup>1</sup><http://ecenter.colorado.edu/images/stories/greeningCU/SustainableCU/SCU2009-10/topographic%20campus%20map.pdf>

<sup>2</sup><http://ecenter.colorado.edu/greening-cu/blueprint-for-a-green-campus>

<sup>3</sup><http://www.colorado.edu/UROP/>

<sup>4</sup><http://www.merrick.com/index.php/geospatial/services-gss/>

<sup>5</sup><http://www.ncalm.cive.uh.edu>

<sup>6</sup><http://www.sanborn.com/>

<sup>7</sup><http://criticalzone.org/>

<sup>8</sup><http://czo.colorado.edu/>

and sharing the level-zero processing steps they took to produce the data. Furthermore, in terms of data capture, NCALM was willing to capture a larger area of interest, allowing us to include the Flatirons in our data set for future shadow modeling. While Merrick did offer to acquire image data of campus, image data was not strictly needed for the project, and would only be useful as visual check on the modeling process. After acquiring the LiDAR data, a data exchange was facilitated with a graduate student researcher for 2009 six-inch National Agriculture Imagery Program<sup>9</sup> (NAIP) data of Boulder, providing imagery for visual checks absent the commercial Merrick imagery.

In evaluating data providers, there was a preference to have less post-processing done by the contractor, instead performing the post-processing of data in-house. This allowed the project to both invest in the community by hiring student research assistants over the summer and fall, and to facilitate a deeper learning experience by using open-source community resources and common software tools to solve complex problems while training project workers in the post-processing steps.

Working with NCALM for the data acquisition allowed for a great deal of insight into the acquisition process. While Merrick had simply asked what the final data product requirements were, NCALM was more inclusive in the planning process, allowing customization of the Pulse Repetition Frequency (PRF), and offering to use a waveform digitizer if needed for the research goals of the project. The waveform digitizer was not requested as it was not necessary for the research goal of the project, and there was no easily accessible way to interpret individual waveforms for hundreds of millions of points. In choosing the Pulse Repetition Frequency, there were inherent trade-offs to consider. A higher PRF would allow for a greater point density at the cost of per-point precision in terms of both elevation accuracy (see Figure 1)<sup>10</sup> and edge detection.

The direct inverse relationship between point density and precision is due to two related factors. The first factor is that the same energy is used at all PRFs, meaning that a higher PRF results in a lower energy per-pulse. Consequently, the laser shot is not quite as tight and the ground size of the laser shot is larger with a higher PRF. As the recorded location of a laser shot is a point (rather than an illuminated

Table 1: Pulse Repetition Frequency vs. Height Accuracy and Point Density

PRF (kHz)	Height Sigma (m)	Points per meter <sup>2</sup>
70	0.033	4.68
100	0.053	6.66
125	0.070	8.78

area), having a smaller area for an averaged positional location is more precise, as the averaged height and positional location is taken over a smaller area. While some of the lost accuracy can be compensated for by statistical analysis (the larger data set would yield results of a greater power), there is still a net loss in precision for urban features. A higher density of points would produce more accurate and precise results in the derivation of bare earth models for heavily forested areas, due to a higher number of points achieving canopy penetration. Given the goals of the project, a PRF of 70kHz was chosen to reflect the urban focus.

The data acquisition flight took place on May 5, 2010. NCALM collected more than 200 million data points over an area that encompassed roughly 40 square kilometers. Of the over 200 million data points, approximately 67 million consisted of the Flatirons, with approximately 134 million covering the main CU campus, East Campus, Bear Creek, South Campus, and areas between and adjacent to university property. A map showing relative density and dispersal of points can be seen in Figure 1. The data was delivered via File Transfer Protocol (FTP) on June 24, 2010 as a set of 67 tiles of binary .las files. The data tiles

<sup>9</sup><http://www.apfo.usda.gov/FSA/apfoapp?area=home&subject=prog&topic=nai>

<sup>10</sup>Data table provided by Juan Carlos Fernandez, NCALM Mission Planner / Laser Operator / Electronics Engineer and University of Houston (UH) Senior Researcher

consisted of per-point attributes including the return number, number of returns per given shot, intensity of return signal, classification, and positional X-, Y- and Z- coordinate information given as Zone 13 Northern Hemisphere UTM coordinates. The return information was collected such that the the first three returns were recorded as separate points, with the fourth return point reflecting either a fourth return or, in cases of greater than four returns, reflecting the position of the last recorded return. The classification categories given were Ground, Unclassified, Medium Vegetation, Building, Low Point (noise), and Water. Additional per-point information not used for this project, but nonetheless included in the data set, were unique point identifiers, a point timestamp, and sensor scan angle.

### 1.3 Project Resources

Since the project’s largest source of funding came from a grant that was maintained by student fees, there was a strong impetus to give back to the student community by hiring student workers. This was reinforced by the philosophical decision to avoid using a commercial LiDAR application such as MARS or Terrasolid<sup>11</sup>, instead investing the saved capital in student training and assistant hours. While five students were initially hired, three of those students eventually lost either interest or availability to help with the project. Of the other two students hired, Trevor Taylor worked on web development while Scott McLamb worked on data processing and analysis. While the web developer’s work does not relate to the methodology and error analysis that is the focus of this thesis, Scott McLamb’s work on data preprocessing and analysis had a direct contribution to the work described here, and I would like to acknowledge his help in processing and creating the 192 final data tiles from which the final results of this thesis are derived.

In keeping with the goals and philosophy of the project, a number of open-source tools were used for data processing and visualization. Martin Isenburg’s NSF-funded work on LiDAR processing (LASTools<sup>12</sup>) was heavily used whenever there was a need to work directly with the point data primitives, and specifically enabled the conversion of LiDAR data from binary files to text files. Python and the associated packages numpy, scipy, pytables and matplotlib were used in developing methods for subsetting data points, and statistical queries. Mayavi<sup>13</sup>, in conjunction with python, allowed visualization of the point primitives.

The single most useful open source project was GRASS GIS<sup>14</sup>, which was helpful in generating rasters of slope, aspect, insolation time, angle of incidence, irradiance, and bare earth models. GRASS was also used for statistical analysis and three-dimensional visualization of raster data, and was used as our general GIS operating environment. While ArcGIS<sup>15</sup> is a more traditional choice for a GIS, GRASS was ultimately chosen instead for several practical reasons. These reasons included support for larger data sets<sup>16</sup>, stronger three-dimensional modeling capabilities with Mayavi and NVIZ, better scripting integration, and a very robust solar insolation module.

The processing work was done on a single server running Linux and hosted in the Meridian Geospatial Visualization Lab<sup>17</sup>. In addition to the aforementioned open-source tools, we also used the CU developed commercial image processing package ENVI for generating canopy DEMs. While generating canopy DEMs is possible in GRASS, ENVI is able to produce the DEMs much faster.

---

<sup>11</sup><http://www.terrasolid.fi/>

<sup>12</sup><http://www.cs.unc.edu/~isenburg/lastools/>

<sup>13</sup><http://code.enthought.com/projects/mayavi/>

<sup>14</sup><http://grass.fbk.eu/>

<sup>15</sup><http://www.esri.com/products/index.html>

<sup>16</sup>GRASS is a 64-bit application, which allows it to allocate more memory than ArcGIS, which as a 32-bit application is limited to a maximum of 4GB of system memory for data analysis.

<sup>17</sup><http://greenwich.colorado.edu>

## 2 Methodology

The methodology for this project was developed and tested on a small, subsetting portion of campus prior to being run for all of Main and East Campus, and Bear Creek. The subset chosen was one square kilometer that included Folsom Stadium, the Engineering Center, the planetarium and other adjacent campus buildings. In describing the production methodology that was chosen, almost all figures will reference this smaller data set (Figure 1).

### 2.1 Development of a Work Flow

The original goals of the project in developing a work flow were to use only the LiDAR data for analysis, and to use existing data sets of campus only for cross-validation or accuracy assessment. The proposed work flow was as follows:

1. Develop a rudimentary classification scheme based off of point primitives to identify grass cover (i.e., ground)
2. Generate bare earth models from classified data
3. Generate height models from bare earth rasters
4. Use height as a strong classifier to identify and classify points associated with buildings
5. Generate vector models of buildings and segment roofs to sections of constant angle and slope
6. Generate solar estimates for roof segments based on slope and aspect

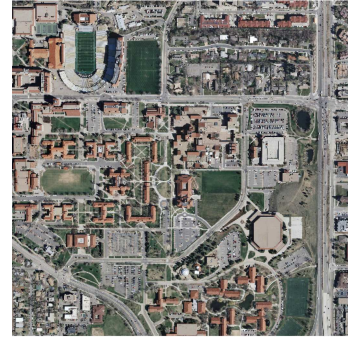
This ambitious approach had several appealing features. Practically speaking, it relied only on the LiDAR data and did not assume any *a priori* knowledge of building locations or features. Programmatically, by limiting solar calculations to extracted roofs, the data set would be substantially reduced.

After attempting to put the above work flow into practice, a number of large limitations and problems became apparent. First, the work flow was overly ambitious; the classification of per point categories, the derivation of both height and bare earth models, the modeling of buildings from point data to three dimensional vector planes— any single task would have been appropriate for an honors thesis on its own. Secondly, the work flow had components that were unnecessary, as vector data sets of building footprints already existed and NCALM had already done a rudimentary point classification. After attempting to generate vector models from point data, it was also apparent that the existing vector data sets of campus were more accurate, as they were derived from building plans. A mid-summer evaluation of the work flow showed it to be needlessly complex, broad and inaccurate.

Due to these difficulties, the work flow was modified to fit a narrower and better-defined scope using a methodology that was both simpler and more accurate. The new work flow, described below, was achieved by moving to a simpler raster data model and by being more methodologically open to using existing data sets:

1. Generate canopy DEMs of the area of interest
2. Generate slope and aspect rasters from the canopy DEMs

Figure 2: Subset of Campus



3. Derive solar estimates from slope and aspect rasters
4. Extract building roofs from solar insolation rasters using existing campus vector data sets

While the generation of insolation rasters is a complicated, multistage process, which will be addressed in the next section, it is important to note some features of the modified work flow.

The use of canopy DEMs, instead of bare earth surface models, simplified the work flow immensely. No classification of points based on primitives is necessary for selection of which points to interpolate; rather, points can be selected on the basis of the existent return number, where all points that are first return points consist of the canopy. These points can then be interpolated with a simple bilinear interpolation routine. This contrasts with the case of developing a bare earth model, where all bare earth points will be last return points, but not all last return points are appropriate to be classified as bare earth. The additional filtering that is required to remove last return points that are not bare earth not only introduces additional classification work, but also introduces greater variability in the density of data points which need to be interpolated. In turn, this variable density limits which interpolation methods can be used on the subsetted points, as both bi-linear interpolation and splines do not perform well for irregularly spaced data with large gaps.

In addition to simpler processing, using a canopy DEM for solar insolation estimates provides more accuracy than only running insolation estimates for individual roof planes. This is because a canopy DEM takes into account all locale sources of shading. For the CU campus, both buildings and trees may shade out building surfaces at certain times of day. By using a canopy DEM instead of just building models, the addition of tree shading gives a more accurate picture of solar insolation.

While there are several methods for generating canopy DEMs that were considered, for our purposes we ultimately decided to use ENVI for generating canopy DEMs. The reasoning was that while GRASS can generate rasters from ASCII points, using ENVI was faster as it allowed generation of DEMs directly from the LiDAR .las binary files, skipping a step of filtering and converting the data to text files that would be necessary for deriving DEMs in GRASS.

## 2.2 Insolation Work Flow and Results

Once canopy DEMs were generated, the insolation work flow could begin. Derivation of Insolation was accomplished using the GRASS module *r.sun*, which could be run both for a single time and for an entire day. The calculation of sun position based off of a given local time, day of the year, and latitude was accomplished by a related GRASS module *r.sunmask*, which could be either run either by itself, or called as part of the *r.sun* module. The *r.sunmask* module uses the SOLPOS 2.0<sup>18</sup> (SOLar POSition and intensity) algorithm developed and maintained by the National Renewable Energy Laboratory<sup>19</sup> (NREL). As the *r.sun* module calls *r.sunmask*, both are described in explaining the work flow.

### 2.2.1 Computational Compromises

Throughout the course of this project, several computational compromises were made. While one of the primary draws of the GRASS GIS program over ArcGIS was its ability to handle data sets larger than four gigabytes, and using GRASS did allow the running of larger mosaicked data set, doing so was not always practical. Doubling the side of a data tile from a kilometer to two kilometers would increase the memory

<sup>18</sup><http://rredc.nrel.gov/solar/codesandalgorithms/solpos/>

<sup>19</sup><http://www.nrel.gov/>



requirements by four times the original tiles requirements, and assuming that a small local window was being used, would also quadruple the processing time. This increase in memory and processing time was expected and acceptable, as the data processed was four times the original tile making the computational time equivalent for tiled and non-tiled approaches. While equivalent in computational time, the large memory overhead would limit the number of concurrent computations to only one large tile at a time, as opposed to a smaller data sets, which could be run four at a time for the same memory and finish in less time. While accuracy was an overriding primary concern when choosing time step sizes, raster cell sizes, and raster extents, there was a conscious effort to choose these parameters in such a way that multiple modules would be able to run simultaneously. These compromises would ultimately allow experimentation and methodological exploration to occur at a faster pace.

### 2.2.2 Basic inputs: Canopy DEMs, Rasters of Slope and Aspect

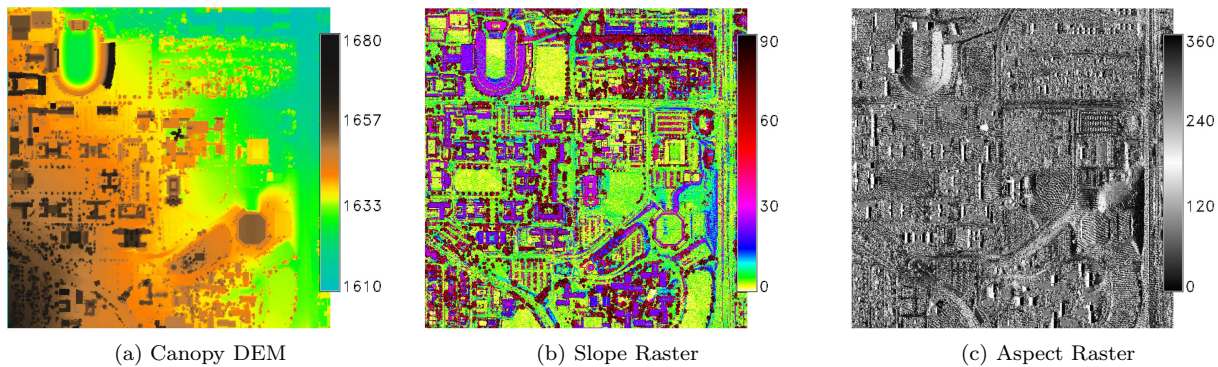


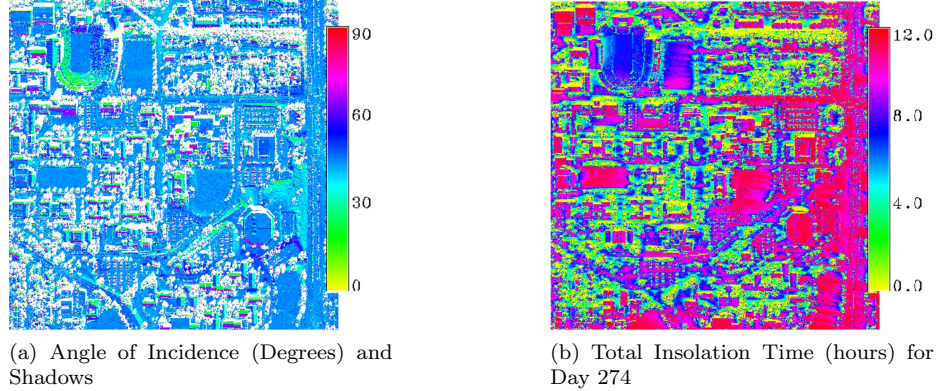
Figure 3: Input Raster Tiles

Angle of incidence rasters are created from a canopy DEM, a slope raster, an aspect raster and sun position parameters. The average point density of the source LiDAR was five points per meter, allowing the generation of a 0.25-meter resolution DEM that on average had one data point per cell. Both the slope and aspect rasters are created from the surface DEM using *r.slope.aspect*, which uses a 3x3 moving window. Figure 2 shows raster tiles from the sample data set for the surface DEM and derived slope and aspect products.

### 2.2.3 Angle of Incidence, Shadows and Insolation Time

The derivation of angle of incidence to the sun for a ground cell is accomplished by taking a given position of the sun and calculating a line of sight to cell, noting the angle of intersection with the ground. In order to accomplish this, the slope and aspect of the ground cell is needed to calculate to ground intersection, while the DEM elevation information is needed to determine line of sight. The position of the sun is given by its direction of origin measured in degrees offset from north (azimuth), and by its angular height in the sky above that bearing, measured in degrees of the sun above the horizon (altitude). Altitude and azimuth can be given explicitly, or can be generated from *r.sunmask* by specifying a complete time and location. To give a complete time, both the day of the year and the Greenwich standard time must be given. While only latitude is needed for a final positional correction of the sun relative to the ground cells, specifying a

Figure 4: Angle of Incidence and Insolation Time



longitude allows for *r.sunmask* to return a sunrise and sunset time.<sup>20</sup> Figure 4a shows an example of a angle of incidence map for day 274 at 1PM and an insolation time map for the same day.

The incidence map is colored with cool to hot colors (blue to red), representing a range from just above a zero-degree angle of incidence to a 90-degree angle of incidence. Since there are areas of the incidence surface that do not have a direct line of sight to sun for the given time, they have no value to represent and are given a null value which corresponds to the color white. Thus, the white areas in the map show the shadows for the given time of day. When running *r.sun* to derive total insolation time for a particular day, the insolation time is calculated by making incidence rasters at a specified time interval, then number of total rasters to be made are determined by sunrise and sunset times for that given day. These incidence angle rasters are then used to calculate binary rasters, where an existing incidence angle at a cell is assigned '1', and where null value cells are assigned '0'. These binary rasters are added together and then multiplied by the time interval to give insolation time rasters as in Figure 4b. Hence, the resolution (both radiometric and spatial) of the insolation time rasters are determined by the temporal step size. For Figure 4b, a step size of 30 minutes was used, resulting in raster cell values of non-shadowed daylight being reported in half hour increments. This also produces the stair-step effect visible on the east side of Folsom Stadium; using a smaller time increment would reduce this effect and given smoother results.

#### 2.2.4 Total, Direct Beam, Diffuse and Reflected Solar Insolation

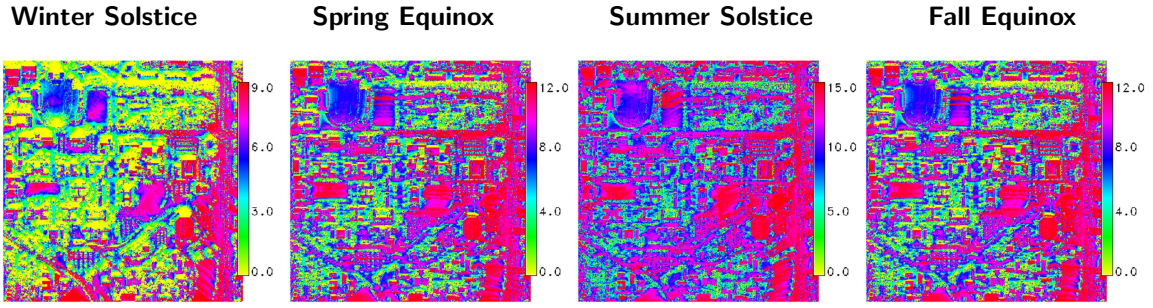
A steradian, or solid angle, from the sun has a (relatively) fixed amount of radiant energy per given unit time. Moving further from the sun increases the area that is needed to fill a given solid angle, while keeping the total radiant output for the steradian constant; thus for two objects of equal size at different distances from the sun, there will be different amounts of incident radiation present as the objects will be exposed to different angular exposure. Since a fixed area of ground surface that is 90 degrees incident to the sun will be exposed to more steradians than a surface that is less than 90 degrees incident, and since energy per steradian is relatively fixed, the angle of incidence to the sun is strongly correlated with the total energy that a surface will receive from the sun.

For direct beam insolation calculations, calculations are accomplished by simply combining incidence, insolation time, and an atmospheric constant. Incoming solar radiation is calculated at 1367 watts per

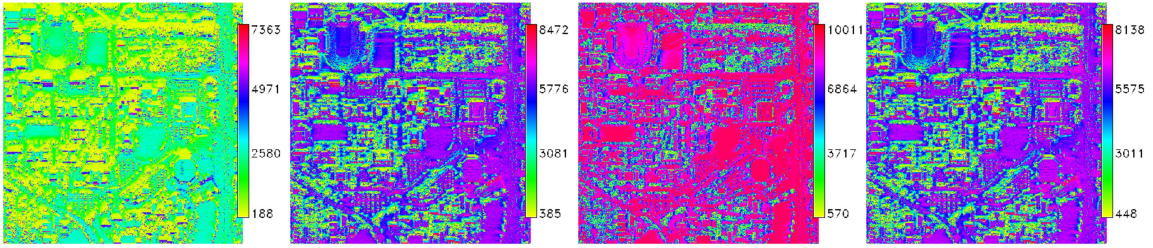
<sup>20</sup>The SOLPOS algorithm is quite robust, correcting for atmospheric refraction— however, the sunrise and sunset times are calculated differently.



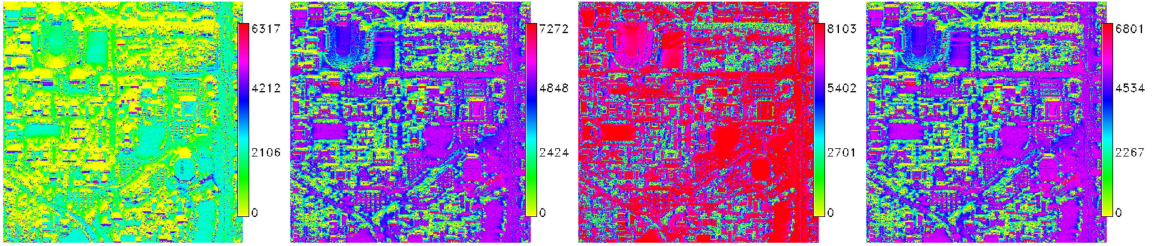
Figure 5: Insolation Raster Tiles



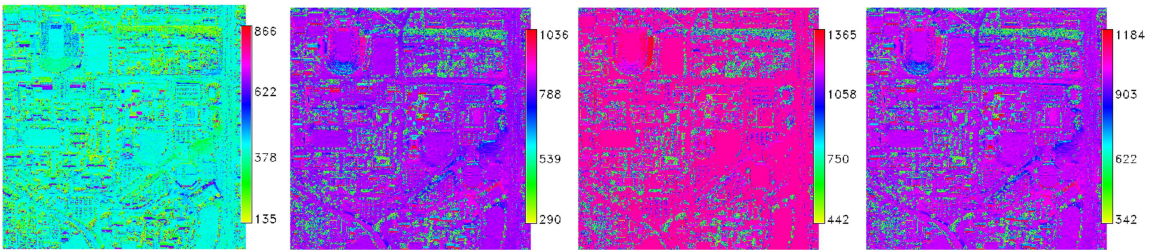
(a) Insolation Time (Hour of Daylight per Day)



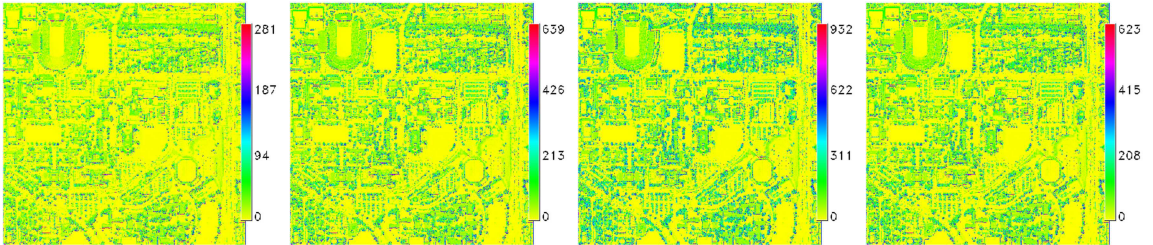
(b) Total Insolation (Watt Hours per Day per Meter Squared)



(c) Direct Beam Insolation (Watt Hours per Day per Meter Squared)



(d) Diffuse Insolation (Watt Hours per Day per Meter Squared)



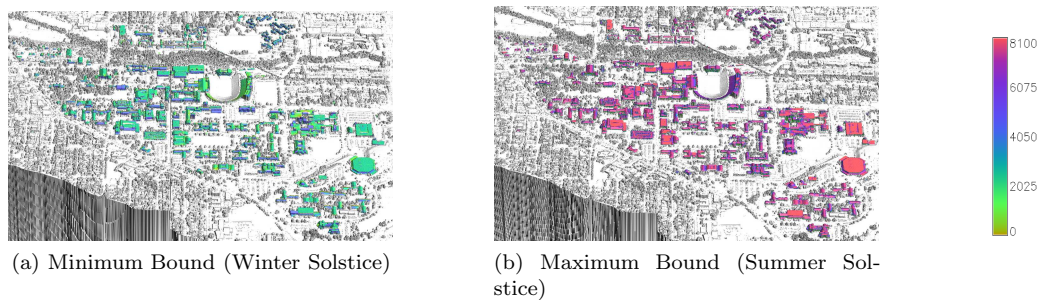
(e) Reflected Insolation (Watt Hours per Day per Meter Squared)

meter squared when ignoring atmospheric effects and hitting the ground surface at a 90 degree angle; the atmospheric constant and the incident angle raster are used to scale down incoming solar radiation to an appropriate level based off of angular dispersion and optical thickness. Total direct beam watt hours per meter for a given day are calculated in modified manner to daily insolation time estimates. The atmospheric constant is used to scale the incoming solar radiation constant for an entire scene, and the sequence of incidence angle rasters are used to scale the surface energy interactions on a per-cell basis. Shadowed areas are given a value of zero, and the sequence of tiles are then used to approximate over the interval using a Riemann sum approximation.

Reflected energy is calculated by applying an albedo factor to the ground. While this can, and should, be done pixel by pixel, it may also be calculated using a single constant, as is the case for Figure 4e. Diffuse radiation refers to energy that is present outside of the direct beam path due to scattering that occurs on both reflected and incoming direct beam radiation. Total energy is the sum of the diffuse, reflected and direct insolation.

### 2.2.5 Final Processing

Figure 6: Merged Direct Beam Insolation Estimates, Extracted and Draped over 0.25 Meter DEM



The final processing steps included the mosiaking of the data tiles, the extraction of rooftops from campus vector data, and the merging of insolation products with other data sets to create visuals. The draped insolation estimates in Figure 6 show watt-hours per meter squared at both the maximum and minimum direct beam irradiation for campus buildings. This represents the current state of the project, which is just beginning to quantify general data about solar insolation into more specific estimates relevant to campus planners. Further plans for analysis (more 'final processing') are discussed in the last section of this paper as well.

## 3 Sources of Error

The insolation rasters are estimates of expected actual solar irradiation which have been modeled from multiple inputs. While any error present in the source input will propagate to estimated output, a distinction can be made between systemic errors of known bound occurring within the source LiDAR, and errors introduced by the processing work flow. This section deals primarily with the latter, and separates work flow error into accounted sources and unaccounted sources of error.



### 3.1 Accounted Sources of Error

Accounted sources of error are errors that are introduced by processing constants. These processing constants are themselves estimates of varying accuracy which could be improved with further modeling and analysis. While the processing constants are estimates, they are bounded estimates. This is in contrast to unaccounted sources of error, which are recognized and acknowledged sources of error that are not explicitly considered modeled in a quantifiable fashion within the analysis of solar insolation.

#### 3.1.1 Atmospheric Optical Thickness

Atmospheric optical thickness is an input parameter of *r.sun* in GRASS. It is expressed as Linke Turbidity Factor, and accounts for optical scattering and dispersion that occurs to radiation between the top of the atmosphere and the ground. Error in estimates of atmospheric optical thickness occurred in two ways: in the specificity of the estimate, and in the estimate itself. While there are modules to derive Linke Turbidity Factor rasters from remotely sensed data, the derivation was outside the scope of this project, given its timeline. The *r.sun* module will accept either a single value for Linke Turbidity Factor, or a raster model with cell by cell values. Unfortunately, we were unable to find either an appropriately scaled Linke Turbidity Factor raster, or even a reliable single local value to use. Thus, error was introduced both in the use of a single value for Linke Turbidity Factor value and in the use of a bounded general estimate, as opposed to a locally derived estimate.

While unable to find precise Linke Turbidity Factor estimates for Boulder, the GRASS documentation did have suggested values for Linke Turbidity Factors in the northern hemisphere at a mild climate based on time of year and general type of location. These suggested values are shown in Table 2 below.

Table 2: Suggested Linke Turbidity Factor Values

Type of Region	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Annual
Mountains	1.5	1.6	1.8	1.9	2.0	2.3	2.3	2.3	2.1	1.8	1.6	1.5	1.90
Rural	2.1	2.2	2.5	2.9	3.2	3.4	3.5	3.3	2.9	2.6	2.3	2.2	2.75
City	3.1	3.2	3.5	4.0	4.2	4.3	4.4	4.3	4.0	3.6	3.3	3.1	3.75
Industrial	4.1	4.3	4.7	5.3	5.5	5.7	5.8	5.7	5.3	4.9	4.5	4.2	5.00

The seasonal variability that is shown above is a function of temperature, which affects both air density and relative humidity, both of which are determinate of scattering and atmospheric optical thickness. The categories modify the Linke Turbidity Factor primarily based on expected pollution, although the mountain category is affected by less total air mass and a lower at-ground air density. In assigning Linke Turbidity Factor values for Boulder, an average was taken between the mountain values and the city values and was then rounded according to how close to the beginning or end of a month the day of year fell. Table 3 shows chosen values for the solstices and equinoxes.

Table 3: Final Linke Turbidity Factor Values

Winter Solstice	Spring Equinox	Summer Solstice	Fall Equinox
2.2	2.6	3.0	3.0

### 3.1.2 Reflected and Diffuse Insolation Estimates

In estimating reflected and diffuse solar insolation, *r.sun* used an albedo constant combined with slope, aspect and elevation. Just as with Linke Turbidity Factor, albedo could be provided either as a single, region-wide constant, or as a cell-by-cell value. A single default value of 0.2 was used to account for albedo while running models. In contrast to the Linke Turbidity Factor, more modeling error is likely to be present due to a lack of specificity because of the more variable nature of ground albedo. While we would expect a relatively stable, continuous gradient when accounting for the Linke Turbidity Factor, ground albedo can be both highly variable and highly discontinuous. A simple example of this can be seen in grass that reaches to the edge of a pond; the grass will reflect heavily in the near-IR while the water will be absorptive across the entire spectrum. While this source of error is more significant than the Linke turbidity factor, it only affects reflected, diffuse and total estimates of insolation, leaving direct beam insolation unaffected. This source of error is also much simpler to correct for in future studies, and a process of correction is discussed at the close of this paper.

While total solar insolation is useful for estimating thermal solar energy production, direct beam insolation is a more appropriate measure of photovoltaic potential. This is because reflected and diffuse light will be proportionally of lower energy, as re-emitted and reflected electromagnetic energy occurs at lower wavelengths. Diffuse and reflected estimates are also affected by an unaccounted source of error, being the permeability of surfaces. This unaccounted source of error, briefly mentioned here for completeness, would cause underestimates of diffuse insolation in areas surrounding trees.

### 3.1.3 Temporal Step Size

In generating solar insolation estimates, there are two temporal step sizes that are used. The first temporal step size is how many days of the year are modeled. By modeling the solstices, boundary conditions are established; the insolation output for the remaining 362 days of the year will, per day, be somewhere between the maximum established at the summer solstice and the minimum established at the winter solstice. Adding the equinoxes provides values for inflection points, while the addition of more days increases the power of a Simpson's, trapezoidal, or midpoint rule calculus estimation of yearly energy potential. Running models for every day of the year will give the most accurate yearly estimate, and by running less than daily models, error is increased.

The same principle of error reduction applies to the temporal step size that is used for generating a single-day estimate of solar insolation. Estimates of insolation time and per-pixel direct, diffuse, and reflected solar energy are generated from Simpson's rule approximations that use intermediate rasters, the number of which is based on the step size at which the rasters are generated. While step sizes as low as one minute can be used, for reasons discussed in Section 2.2.1, a step size of 30 minutes was chosen for generating solar models. While this does increase the error of the insolation estimates compared to a step size of one, five, or 10 minutes, there are two benefits realized beyond reduced computational time that is achieved with a step size of 30 minutes.

The first benefit to using a larger step size relates to problems within the GRASS module at times close to sunrise and sunset. When calculating the angle of the sun above the horizon, the SOLPOS algorithm used by GRASS accounts for atmospheric refraction; however, the module that calculates sunrise and sunset times does *not* account for atmospheric refraction, causing the module not to run even if the sun has a positive horizon angle. While this can be fixed by scripting, allowing a larger step size avoids the issue entirely. An

appropriate compromise for running future models would be a step size of four or five minutes, which would produce smoother results than a half hour while still avoiding sunrise and sunset issues. A step size of four or five minutes would also be either four or five times computationally faster than running a minute interval step size.

The second benefit to using a large step size relates to the unaccounted error of non-local shading discussed in the next section. The solar insolation time increment of a half hour masks the large variability occurring due to the Flatirons, which cause disparate shading times of 15-20 minutes throughout campus. While the larger step size chosen does alleviate the task of including shading from the Flatirons, a more robust future study would reduce the temporal step size and address the shading due to the Flatirons explicitly.

## **3.2 Unaccounted Sources of Error**

### **3.2.1 Non-local Shading**

One of the computational compromises made during this exploratory round of data processing was the decision not to include the shading of the Flatirons. While tiling is very effective for dealing with large data sets, the tiled shadow maps of Boulder Valley were predicated on the assumption that tiling was alright, given that most shading was local and would not cross tile boundaries. This assumption clearly does not hold for the Flatirons, which are more prominent by several orders of magnitude in shadow length than Folsom Stadium, the physics towers, or any other prominent structure on campus. Simply put, there was not enough memory available to include the Flatirons at the resolution at which Boulder Valley was being processed.

In looking to future work, there are three solutions to enable processing of the Flatiron shadow effects: use different resolution rasters, get more memory, or use a horizon angle raster to give cell-by-cell horizon information for the area of interest. This last method is the most promising, but additional research is needed to determine if this interferes with generation of locale shadow relief.

### **3.2.2 Clouds and Local Weather**

While insolation was explicitly accounted for, cloud coverage was not. The complexity and variability of the local weather system precluded any predictive analysis of cloud coverage for this project. While predictive modeling is out of the question for this or future analysis, one approach that is promising to at least partially account for weather is to look at historical records. While taking the oft-mentioned statistic that Colorado has 300 sunny days a year gives a very rough idea of cloud cover, a more thorough analysis would include looking at monthly historical weather patterns to adjust hours of sunshine appropriately. Historical records could also form the basis of a stochastic model for further studies, should the interest or need arise.

### **3.2.3 Tiling Effects**

One unaddressed source of error encountered during the analysis is the effect of tiled data. While the error due to tiling was small in magnitude due to the large (4000x4000) rasters that were used, it was a discernible and appreciable effect. Even though tiling effects were limited to a single boundary-edge-pixel per adjacent tile present when calculating slope and aspect, there were more noticeable effects present in shadow maps where a building shadow would be cut by the edge of a tile. While edge effects could be mitigated by doing adjacent neighborhood analysis, a more vigorous approach would be to calculate the edge boundaries explicitly with re-sampled data. The easiest way to accomplish this would be with thin rasters that covered

only the edges, but a slightly more complex and preferable solution would be to enlarge the tiles to an overlapping size, and trim the excess edge when not needed. The base DEM was tiled but continuous, with no edge effects or lost pixels; given a similar set up, it would be simple to re-sample to a larger raster size (4050x4050 or 4100x4100), and remove the edges after the insolation data processing was finished and data tiles were to be re-merged again.

## 4 Concluding Remarks and Future Work

While explicit yearly insolation estimates have not yet been completed, the goal of this honors thesis has been to examine the methodology that produces the data from which these estimates will be derived. When these estimates are made explicit, it will be either as a result of Riemann sum estimation from a larger subset of insolation days, or by running daily solar models using the methodology and best practices described in this thesis. In either case, the accuracy of the estimates that will be calculated in the coming months will ultimately depend on the robustness of the method for generating the underlying source data.

### 4.1 Correction of Errors

In light of the errors discussed in the previous section, several best practices for further solar mapping are noted here for future studies. As stated above, resampling to larger overlapping tiles would reduce edge effects, as would using a smaller step size for time increments in *r.sun*—specifically, extending raster tiles a quarter percent and using a step size of four or five minutes. While the step size would increase processing time, using half-meter raster cells would increase raster operations by a factor of four, and would possibly allow tiles to be batched in groups of six instead of four, providing a reasonable balance. Taking advantage of an increased temporal step size would also make it appropriate to account for the Flatirons through a horizon angle raster, or at the very least, by a sunset modification time.

Reflected, diffuse, and total insolation estimates would be greatly enhanced by making albedo rasters. These could be produced from the intensity band of the LiDAR or by combining RGB brightness values with LiDAR intensity to calculate luminance. While this would not affect beam estimates, I believe providing better albedo estimates would be one of the single largest sources of increased certainty for the other insolation estimates. The Linke Turbidity Factor could also be improved on, possibly by making a turbidity raster. However, the most pressing and immediate work flow changes— that is, changes that should be implemented prior to running the next three month batch of data tiles— are to implement an albedo raster, re-sample to a greater extent at a lower resolution, and change step sizes to reflect greater parallel processing and produce smoother results. It is my goal to implement these changes and the general work flow as a script to run until May 2011.

### 4.2 Other Work

For work to be finished beyond the scope of this thesis, there is both more project work and more pure research to be conducted on the LiDAR data set. With regards to producing solar estimates, there are the

Figure 7: DEM, NAIP, and Insolation





additional steps of comparing energy potential with power consumption, adjusting for conversion efficiencies to accurately gauge energy potential, and calculating roof space needed for a given reduction in carbon power.

In terms of pure research, I would still like improve on producing bare models. As part of the initial work flow that was later abandoned, some basic bare earth models were created using simple classification and Inverse Distance Weighting (IDW). For future bare earth modeling, both kriging and natural neighbor interpolation would be of comparative interest. A large reason for deriving bare earth models is the door that it opens to height mapping and classification. Finally, encoding per-point attribute data as tables in a database would greatly increase the power of future work with the data set, and is a long-term future goal for work with this project.